International Journal of Scholarly Research (IJSR)
Vol-1, Issue-1, 2017
http://www.scholars.org.in

"Review on an Efficient Approach for Ranking Based Fraud Detection in Android Mobile Market"

Sumit Annaji Kale [1], Prof.Naziya Pathan [2]

[1] Departement of computer science and engg,
Nuva College of Engineering and Technology, Nagpur
Sumit.kale28@gmail.com

[2]Departement of computer science and engg,
Nuva College of Engineering and Technology, Nagpur

Abstract: Ranking fraud in the mobile App market refers to fraudulent or unrepresentative activities which have a purpose of knocking up the Apps in the popularity list or make ourselves famous. Without a doubt, it becomes more and more numerous for App developers to use sheltered means, such as inflating their Apps' sales or posting phony App ratings, to execute ranking fraud. While the significance of preventing ranking fraud has been extensively predictable, there is limited consideration and research in this area. In this Paper we are reviewing various a holistic view of ranking fraud and propose a ranking fraud detection system for mobile Apps. Specifically we first study various ranking fraud. Moreover, we examine different methodologies and characterised it into three types of verification in fraud detection, i.e., ranking based evidences, rating based evidences and review based evidences, by representing Apps' ranking, rating and review behaviours through statistical hypotheses tests. In addition, we will also suggest an optimization based combination method to incorporate all the evidences for fraud recognition.

Keyword: Mobile Apps, ranking fraud detection, evidence aggregation, historical ranking records, rating and review.

1. INTRODUCTION

The number of mobile Apps has grown at a overwhelming rate over the past few years. For example, as of the end of April 2013, there are more than 1.6 million Apps at Apple App store and Google Play. To motivate the development of mobile Apps, many App stores commence daily App head boards, which exhibit the graphic representation rankings of most accepted popular Apps. Without a doubt, the App leader board is one of the most significant ways for subsidise mobile Apps. A privileged rank on the leader board frequently leads to a enormous number of downloads and million dollars in profits. Therefore, App developers be inclined to investigate various ways such as promotion drive to promote their Apps in order to have their Apps ranked as soaring as possible in such App leader boards. However, as a recent inclination, instead of relying on traditional marketing solutions, sheltered App developers remedy to some fraudulent means to intentionally boost their Apps and in due course manipulate the chart rankings on an App store. This is usually put into service by using so-called farms man water armies to inflate the App downloads, ratings and reviews in a very squat time. For example, an article from Venture Beat account that, when an App was sponsored with the help of ranking manipulation, it could be boosted from number 1,800 to the top 25 in Apple top free leader-board and more than 50,000-100,000 new users could be obtained within a couple of days. In fact, such ranking fraud elevates enormous concerns to the mobile App diligence. For example, Apple has notified of cracking down on App developers who perpetrate ranking fraud in the Apple App store. In the literature, while there are some related work done by other developers, such as web ranking spam recognition, online review spam exposure, and mobile App commendation, the difficulty of distinguish ranking fraud for mobile Apps is at rest under-explored. To fill this critical void, in this paper, we propose to build up a ranking fraud exposure structure for mobile Apps. Along this statement, we recognize some very significant confronts that we have to overcome at any cost. First, the thing we have to take in consideration or to focus that is ranking fraud does not always take place in the whole life cycle of an App, so we need to spot the time when fraud occurs. Such challenge can be observed as detecting the local irregularity instead of global irregularity of mobile Apps. Second, as we know there are many number of apps are present in a real world. Due to the huge number of mobile Apps, it is complicated to physically mark ranking fraud for each App, so it is significant to have a scalable way to mechanically discover ranking fraud without using any benchmark information. Finally, due to the energetic nature of chart rankings, it is not easy to categorize and make sure the confirmations linked to ranking fraud, which encourages us to determine some understood fraud patterns of mobile Apps as confirmations. Then, with the investigation of Apps' ranking

behaviours, we find that the fraudulent Apps often have dissimilar ranking patterns in each important session weigh against with normal Apps. Thus, we distinguish some fraud evidences from Apps' chronological ranking records, and build up three functions to extort such ranking based fraud evidences. Nonetheless, the ranking support confirmation can be pretentious by App developers' reputation and some justifiable marketing campaigns, such as "limited-time discount". As a result, it is not adequate to only use ranking based confirmations. Therefore, we further propose two types of fraud confirmations based on Apps' rating and review past data, which reflect some irregularity patterns from Apps' historical rating and review records. In accumulation, we develop an unsubstantiated evidence-aggregation method to incorporate these three types of evidences for estimating the authority of leading sessions from mobile Apps. It is worth noting that all the evidences are extorted by modelling Apps' ranking, rating and review behaviours through arithmetical hypotheses tests. The proposed structure is scalable and can be unmitigated with other domain produced evidences for ranking fraud detection. Finally, we evaluate the proposed system with real-world App data composed from the Apple's App store for a long time period, i.e., more than two years.

BACKGROUND

Web Advertising:

Advertising or promoting on the world wide web is persistent, and allows for services such as websites, search, and email to be make available to customers for free by including advertisements (ads) as part of the substance exhibited to the user or customer. Website owners and other provision contributors (called publishers in advertising jargon) characteristically include ads through a third party called an ad contributor, which handles nodding and decide on advertisements, as well as paying publishers for ads shown to their users. On the Web, this is typically employ as and <iframe> or <script> HTML element implanted in the publisher's webpage, with a src (source attribute) attribute that points to the ad provider's ad server. When the web page is encumbered by a browser, the ad is colonized via an ad demand, which contains the contributors or ad providers ID and information or some personal details about the user that is used to select a appropriate ad (known as targeting information). The ad server proceeds three pieces of substance once an ad is selected: the ad content URL, a click URL, and a pixel URL.

The ad substances is characteristically hosted by the ad provider (usually through a CDN- Content delivery network) instead of the digital marketer who owns the ad, guarantying the content will be violable when the ad is loaded. Marketers those taking efforts to promote their ad's and who are paying for their ads to be circulated by the ad provider want to ensure the ad provider is not fraudulently billing them, so they themselves host or show a tracking pixel (or web bug) that is added by browsers along with the ad so that the marketers can autonomously authenticate that ads are being requested.

Finally, the click URL designates which web page should be opened when a user clicks on an ad. The click URL typically directs to the ad provider's ad server, which records the clicks and then redirects the user to the marketer's landing page which contributor ad server has decided to show to the user of click event. An absolute ad request, response, and display of the ad and pixel to the user are called an impression, and opening the click URL is a click. Publishers are paid based on how many impressions and clicks on URL their content generates or they achieved by their ad's marketing.

Web Ad Fraud:

Unprincipled publishers may initiate their ad revenues by having mechanical bots stopover their website and click on ads. This is referred to as ad fraud (or click fraud), and is a serious security issue as digital marketers who pay to have their ads shown online will not receive any business advantage for ads shown to boost. Although hard numbers on the amount of ad fraud is hard to conclude, unadventurous estimates suggest 10% of Web ad traffic is due to fraud. In order to obtain revenue, fraudsters must remain unobserved while issuing large numbers of ad requests and clicks. To do so, they employ a number of procedure and techniques.

First, the proportion of click requests to requested ads is kept low (around 1%) to avoid doubt, as ads are infrequently clicked on by real users. This means fraudsters concern far more ad requests than click requests. Second, fraudsters do not rely on or dependent on a single publisher account, but relatively have many accounts from many ad contributors which they alternate through while issuing requests. Not only does this moderate the impact or outcome of any single user account being detected or verified, it also declines the enormity of fraudulent requests for each publisher ID and ad provider.

Finally at last, fraudsters use botnets as the bots run code that without fail visits the fraudsters' webpage's in the backdrop and clicks on the ads situated there, so that the fraudsters obtain revenue. Botnets allow fraudsters to stay surreptitious as the bots are real user devices which have been negotiated.

Android App Advertising:

Many Android applications are distributed or allotted for free on app markets, and use ads implanted in the user interface or at the when the user are going to interact with the app of the app to make money for the developer. The

developer must register with an Android ad provider or contributor, which supply the developer with a publisher ID and an ad library to include in their app. The library is conscientious for fetching and displaying ads when the app is being run. Demanding an ad for an app is equivalent to doing so on the web: an ad request is made over HTTP to the ad servers which include the developer's publisher ID and user targeting information or personal detail data.

The ad server returns or delivers the ad's content URL, click URL, and any tracking pixel URLs which must be bring to display the ad. In fact, many ad libraries desire to employ making requests and displaying in fact showing ads simply by loading a traditional HTML ad ingredient in a web view. The primary distinction between web and Android app advertising is that ad libraries are implemented in application code, and often enclose special application-only logic, for example mechanically collecting or gathering user or client's intentional information related data or refreshing the ad or changing the ads.

2. RELATED WORK

As we know before us many great peoples worked on this android app ranking fraud detection through ads so we just go through their study work and take inspiration from their work and build our improved system.

Xiong and Zhu[1] had projected a ranking fraud detection system for android mobile apps. In this paper principally, they both demonstrated that ranking fraud take place in most important sessions for each app from its previous ranking accounts. Then, they recognized ranking based, rating based and review based confirmation for discovering ranking fraud. Moreover they proposed an optimization based aggregation system to merge all the evidences for estimate the consistency of most important sessions from mobile apps.

Priyanjai and Pankaj[2] planned techniques for assessment of investigation and invent pattern of android apps based on cloud computing and data mining. They developed system ASEF and SAAF for android apps to achieve protection. They also explain a tactic that performs apps security and provide user friendly interface on a mobile phone.

Anuja A. Kadam ,Pushpanjali M. Chouragade [3] make available a disciplined study on the different procedures of malicious application recognition in android mobiles. The examination of authorization induces possibility in Android apps on a large-scale in three stages. First upon position all the entity permissions with respect to their feasible risk with different processes. Secondly, classify subsets of risk permissions. Then using several algorithms identifies the suspected apps based on the recognized subsets of risky permissions.

Jakub Zilincan ,Michal Gregus [4] had given the dedicated work on Search engine optimization techniques, often summarized to SEO, should lead to first situation in unprocessed search results. Some optimization techniques or procedures do not modify over time, yet still form the foundation of SEO. However, as the Internet and web design develop enthusiastically, new optimization procedures come in to account and sometime does not work. Thus, they have focused on most important features that can help to get better a pose in search outcome. It is important to accentuate, that none of the procedure can make sure it because search engines have complicated algorithms, which measure the superiority of Web pages and obtain their position in search results from.

Xiang Wang, Yan Jia , Ruhua Chen, Bin Zhou [5]in that they had told users can interpret themselves using free tags in micro-blogging website such as Sina Weibo. The tags of a user exhibit. The description of the user and are normally in a unsystematic direct without any significance or importance information. It restricts the usefulness of user tags in system suggestion and other applications. They also proposed a user tag ranking representation which is based on interactive and attractive dealings between users. Manipulate power between users is measured in our user tag ranking method. Significance scores between tags and users are also utilized to rank user tags.

Young Ihm. Woong-Kee Loh[6] had noticed that App store and android market have knowledgeable a noteworthy growth in terms of app numbers. Since they had exposed 85% of apps during the ranks, it is significant to develop effective and efficient user friendly app ranking examine tools. Woong offered a structure called App Analytic. In this He discovered the correspondence of app ranking data about admired social networking sites. Exclusively, they observed association between various features of social networking sites on Internet and android market. The results of their correlation analysis disclose that there is a strong positive association of the number of app downloads with the number of scheduled users and page rank. They also supply an in-depth examination on the major factors that impact the association.

3. PROPOSED METHOD

1. Mining Leading or primary Sessions

There are two main steps for mining primary sessions. First, we need to determine leading measures from the App's historical previous ranking records. Second, we need to collaborate neighbouring leading events for developing leading sessions. Mainly, Algorithm 1 reveals the pseudo code of mining leading sessions for a given App.

2. EXTORTING EVIDENCES FOR RANKING FRAUD RCOGNIZATION

- **Ranking Based Evidences**

By analyzing the Apps' historical previous ranking accounts, Apps' ranking behaviours in a leading incident

always assure a specific ranking pattern, which consists of three different ranking segments, expanding phase, maintaining phase and collapse phase. Mainly, in each leading event, an App's ranking first improve to a peak or extent position in the leaderboard (i.e., rising phase), then maintain ssuch peak position for a phase (i.e., maintaining phase), and at last declines till the end of the event (i.e., recession phase). Definitely, such a ranking pattern confirms an significant considerate of leading event. In next section we formally describe the three ranking phases of a leading event.

- **Rating Based Evidences**

The ranking based evidences are helpful for ranking fraud recognition. However, sometimes, it is not satisfactory to only use ranking based evidences. Take an example, some Apps formed by the legendary developers, such as Gameloft, may have some leading events with large principles of you due to the developers' trustworthiness and the "word-of-mouth" advertising effect. Moreover, some of the permissible marketing services, such as "limited-time discount", may also consequence in significant ranking based evidences. To solve this matter, we also study how to extort fraud evidences from Apps' historical previous rating records.

- **Review Based Evidences**

Further ratings, most of the App stores also allow users to write some textual comments as App reviews to submit to the developer. Such reviews can reflect the personal observations and usage understanding of breathing users for particular mobile Apps. Indeed, review management is one of the most important viewpoints of App ranking fraud. Specifically, before downloading or purchasing a new mobile App, users often first read its previous historical reviews to simplicity their conclusion making and a mobile App includes more encouraging reviews may attract more users to download. Therefore, imposters often place counterfeit reviews in the leading sessions of a specific App in order to inflate the App downloads, and thus boost the App's ranking position in the leaderboard.

- **Evidence Aggregation or Association**

After extorting three types of fraud evidences, the next dare is how to merge them for ranking fraud detection. Indeed, there are many ranking and evidence association techniques in the literature that we have studied before, such as transformation based models, achieve based models, and Dempster-Shafer rules. However, some of these methods spotlight on learning a worldwide ranking for all contenders.

4. CONCLUSIONS

A ranking fraud detection system for mobile Apps show that ranking fraud take place in most important sessions and make available a method for mining leading sessions for each App from its previous historical ranking records. Then, our study recognizes that it can be generally describes into three sort i.e. Ranking based evidences, rating based evidences and review based evidences for recognizing ranking fraud and an optimization based aggregation method to integrate or make association of all the evidences for evaluating or calculating the trustworthiness of leading sessions from mobile Apps.

5. REFERENCES

[1] Hengshu Zhu, Hui Xiong, Senior Member, IEEE, Yong Ge, and Enhong Chen, Senior Member, IEEE Discovery of Ranking Fraud for Mobile Apps‖ IEEE Transactions On Knowledge And Data Engineering, Vol. 27, No. 1, January 2015.

[2] Pranjali Deshmukh, Pankaj Agarkar ―Mobile Application For Malware Detection‖ International Research Journal of Engineering and Technology (IRJET) e-ISSN: 2395 -0056 Volume: 02 Issue: 02 | May-2015 www.irjet.net

[3] Anuja A. Kadam ,Pushpanjali M. Chouragade ―A Review Paper on: Malicious Application Detection in Android System‖International Journal of Computer Applications (0975 – 8887) National Conference on Recent Trends in Computer Science & Engineering (MEDHA 2015).

[4] Jakub Zilincan ,Michal Gregus "Improving Rank of a Website in Search Resuts – a Experimental Approach"2015 10th International Conference on P2P, Parallel, Grid, Cloud and Internet Computing978-1-4673-9473-4 /15 $31.00 © 2015 IEEE

[5] Xiang Wang,Yan Jia , Ruhua Chen, Bin Zhou , "Ranking User Tags in Micro-blogging Website",978-1-4673-6850-6/15 .2015 IEEE

[6] App Analytic: A Study on Correlation Analysis of App Ranking Data Sun-Young Ihm; Woong-Kee Loh; Young-Ho Park Cloud and Green Computing (CGC), 2013 Third International Conference on Year: 2013 Pages: 561 ‐ 563, DOI: 10.1109/CGC.2013.95 IEEE Conference Publications

[7] Jakub Zilincan ,Michal Gregus "Improving Rank of a Website in Search Resuts – a Experimental Approach"2015 10th International Conference on

P2P, Parallel, Grid, Cloud and Internet Computing978-1-4673-9473-4 /15 $31.00 © 2015 IEEE

[8] L. Azzopardi, M. Girolami, and K. V. Risjbergen, "Investigating the relationship between language model perplexity and ir precision-recall measures," in Proc. 26th Int. Conf. Res. Develop. Inform. Retrieval, 2003, pp. 369–370.

[9] D. M. Blei, A. Y. Ng, and M. I. Jordan, "Latent Dirichlet allocation," J. Mach. Learn. Res., pp. 993–1022, 2003.

[10] Y. Ge, H. Xiong, C. Liu, and Z.-H. Zhou, "A taxi driving fraud detection system," in Proc. IEEE 11th Int. Conf. Data Mining, 2011, pp. 181–190.

[11] D. F. Gleich and L.-h. Lim, "Rank aggregation via nuclear norm minimization," in Proc. 17th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining, 2011, pp. 60–68.

[12] T. L. Griffiths and M. Steyvers, "Finding scientific topics," Proc. Nat. Acad. Sci. USA, vol. 101, pp. 5228–5235, 2004.

[13] G. Heinrich, Parameter estimation for text analysis, " Univ. Leipzig, Leipzig, Germany, Tech. Rep., http://faculty.cs.byu.edu/~ringger/CS601R/papers/Heinrich-GibbsLDA.pdf, 2008.

[14] N. Jindal and B. Liu, "Opinion spam and analysis," in Proc. Int. Conf. Web Search Data Mining, 2008, pp. 219–230.

[15] J. Kivinen and M. K. Warmuth, "Additive versus exponentiated gradient updates for linear prediction," in Proc. 27th Annu. ACMSymp. Theory Comput., 2005, pp. 209–218.

[16] A. Klementiev, D. Roth, and K. Small, "An unsupervised learning algorithm for rank aggregation," in Proc. 18th Eur. Conf. Mach. Learn., 2007, pp. 616–623

[17] E.-P. Lim, V.-A. Nguyen, N. Jindal, B. Liu, and H. W. Lauw, "Detecting product review spammers using rating behaviors," in Proc. 19th ACMInt. Conf. Inform. Knowl. Manage., 2010, pp. 939–948.

[18] Y.-T. Liu, T.-Y. Liu, T. Qin, Z.-M. Ma, and H. Li, "Supervised rank aggregation," in Proc. 16th Int. Conf. World Wide Web, 2007, pp. 481–490.

[19] A. Mukherjee, A. Kumar, B. Liu, J. Wang, M. Hsu, M. Castellanos, and R. Ghosh, "Spotting opinion spammers using behavioral footprints," in Proc. 19th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining, 2013, pp. 632–640.

[20] A. Ntoulas, M. Najork, M. Manasse, and D. Fetterly, "Detecting spam web pages through content analysis," in Proc. 15th Int. Conf. World Wide Web, 2006, pp. 83–92.

[21] G. Shafer, A Mathematical Theory of Evidence. Princeton, NJ, USA: Princeton Univ. Press, 1976.

[22] K. Shi and K. Ali, "Getjar mobile application recommendations with very sparse datasets," in Proc. 18th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining, 2012, pp. 204–212.

[23] N. Spirin and J. Han, "Survey on web spam detection: Principles and algorithms," SIGKDD Explor. Newslett., vol. 13, no. 2, pp. 50–64, May 2012.

[24] M. N. Volkovs and R. S. Zemel, "A flexible generative model for preference aggregation," in Proc. 21st Int. Conf. World Wide Web, 2012, pp. 479–488.